

Classificação de sinais WGC e WGN

Fernando Henrique dos Santos e Marcio Eisencraft

Resumo— O *white Gaussian chaos* (WGC) é um mapa que gera sinais caóticos com a mesma densidade espectral de potência, sequência de autocorrelação, medidas estatísticas de posição e dispersão de um processo ruído branco gaussiano (WGN - *white Gaussian noise*). Neste trabalho utilizam-se redes neurais do tipo perceptron multicamadas para determinar se um dado sinal foi gerado por um WGC ou é um função-amostra de um processo WGN. Explora-se a natureza do sinal WGC para otimizar a arquitetura da rede.

Palavras-Chave— sinais de tempo discreto, caos, ruído, redes neurais.

Abstract— The *white Gaussian chaos* (WGC) is a map that generates chaotic signals with the same power spectral density, autocorrelation sequence, statistical measures of position and dispersion of a *white Gaussian noise* (WGN) process. In this work, multilayer perceptron neural networks are used to determine whether a given signal was generated by a WGC or is a sample function of a WGN process. The nature of the WGC signal is exploited to optimize the network architecture.

Keywords— discrete-time signals, chaos, noise, neural networks.

I. INTRODUÇÃO

O *white Gaussian noise* (WGN) é um processo estocástico obtido a partir de amostras gaussianas independentes. É frequentemente utilizado como ruído nos experimentos numéricos das áreas de ciência e tecnologia para incluir aleatoriedade aos modelos teóricos, tornando-os mais próximos da realidade física [1]. O WGN possui densidade espectral de potência (DEP) constante, ou seja, sua DEP independe da frequência e sua função de autocorrelação é impulsiva.

O *white Gaussian chaos* (WGC) é um mapa que gera sinais caóticos [2], [3], com a mesma DEP, autocorrelação, medidas estatísticas de posição e dispersão de um sinal WGN [4]. Como o sinal gerado pelo WGC é caótico, por definição, tem natureza determinística ao contrário do sinal produzido pelo WGN. Assim as medidas mais tradicionais [5] para se caracterizar um sinal, não são suficientes para classificar um sinal em WGC ou WGN.

Esse trabalho tem como objetivo fazer um estudo dessa classificação por meio de redes neurais do tipo perceptron multicamadas (MLP) [6] e explorar a natureza do sinal WGC para otimizar a arquitetura dessas redes.

O artigo foi dividido da seguinte forma: Na Seção II revisitam-se as equações que geram o mapa WGC e suas propriedades. Na Seção III apresentam-se a arquitetura das redes utilizadas e as características resultantes dessa arquitetura. A seguir, os principais resultados são apresentados na Seção IV.

Fernando Henrique dos Santos, Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil, e-mail: fhsantos@lcs.poli.usp.br; Marcio Eisencraft, Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil, e-mail: marcio@lcs.poli.usp.br.

Por fim, na Seção V discutem-se as conclusões desse estudo e sugerem-se trabalhos futuros.

II. O MAPA WGC

O mapa tenda $\Lambda : (0,1) \rightarrow (0,1)$ é definido como

$$z(n+1) = \Lambda(z(n)) = \begin{cases} 2z(n), & 0 < z < 1/2 \\ 2(1-z(n)), & 1/2 \leq z < 1 \end{cases} \quad (1)$$

em que n é a variável temporal e $z(0) \in (0,1)$ é uma condição inicial dada. As órbitas geradas por esse mapa tem densidade invariante uniforme no intervalo $(0,1)$, expoente de Lyapunov [3] iguais a $\ln 2$ e autocorrelação impulsiva [7].

Definindo $T : (0,1)^2 \rightarrow (0,1)^2$ como aplicação de Λ nas coordenadas de $y = [y_1 \ y_2]^T$, temos

$$y(n+1) = T(y(n)) = [\Lambda(y_1(n)) \ \Lambda(y_2(n))]^T. \quad (2)$$

Foi proposto em [4] um mapa W que ao ser conjugado com T [2], possui uma densidade invariante gaussiana, expoentes de Lyapunov $\ln 2$ e autocorrelação impulsiva.

Seja G a transformação que, aplicada a duas variáveis R_1 e R_2 independentes que possuem função densidade de probabilidade uniforme no intervalo $(0,1)$, gera duas variáveis aleatórias independentes gaussianas de média nula e variância σ^2 . Essa transformação $G(0,1) \rightarrow \mathbb{R}^2$ é dada por

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = G \left(\begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \right) = 2\sigma^2 \begin{bmatrix} \ln\left(\frac{1}{1-R_1}\right) \cos(2\pi R_2) \\ \ln\left(\frac{1}{1-R_1}\right) \cos(2\pi R_1) \end{bmatrix} \quad (3)$$

A transformação G é invertível e sua inversa $G^{-1} : \mathbb{R}^2 \rightarrow (0,1)^2$ é dada por [8]

$$\begin{bmatrix} R_1 \\ R_2 \end{bmatrix} = G^{-1} \left(\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} \right) = \begin{bmatrix} 1 - \exp\left(-\left(\frac{Y_1^2 + Y_2^2}{2\sigma^2}\right)\right) \\ \frac{1}{2\pi} \operatorname{atan2}(Y_2, Y_1) \end{bmatrix} \quad (4)$$

em que $\operatorname{atan2}(x)$ é o arco do 4º quadrante cuja tangente é x .

O mapa WGC $W : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ é então definido como

$$x(n+1) = W(x(n)) \quad (5)$$

com

$$W = G \circ T \circ G^{-1}. \quad (6)$$

Note que, por definição W é conjugado de T e assim apresenta expoentes de Lyapunov $(\ln 2, \ln 2)$.

A Figura 1 apresenta uma comparação em termos de histograma e sequência de autocorrelação de uma órbita gerada pelo mapa W e um sinal WGN.

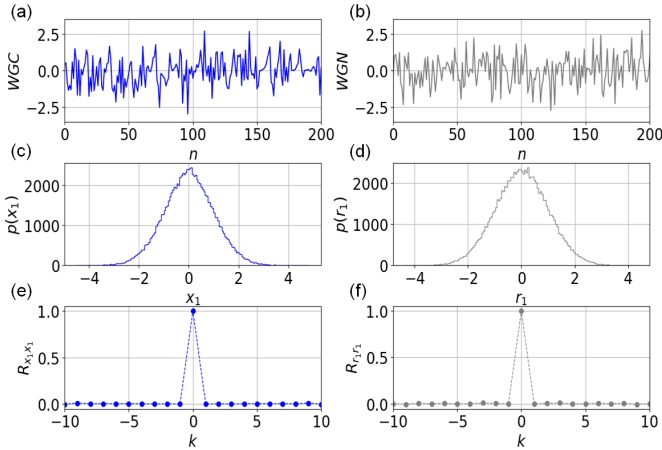


Fig. 1. (a) trecho de uma das coordenadas de órbita gerada pelo mapa W com condição inicial aleatória e $\sigma^2 = 1$; (b) função-amostra de um processo WGN com $\sigma^2 = 1$; (c) e (d) função densidade de probabilidade das órbitas do mapa WGC e do WGN, respectivamente; (e) e (f) seqüências de autocorrelação para ambos os sinais.

III. CLASSIFICANDO SINAIS WGC E WGN

Inspirados no modo como o cérebro aprende, armazena informações, as processa e depois as reconhece, cientistas tem projetado sistemas capazes de executar tarefas específicas, como reconhecer objetos em uma imagem [9], emoções de uma pessoa, por meio de fala ou escrita [10] e realizar previsões de séries temporais [11].

Um desses sistemas é a rede neural, que assim como o cérebro, possui unidades básicas de processamento (neurônios) que quando conectadas entre si, formam estruturas capazes de armazenar conhecimento. As redes neurais utilizadas nesse trabalho são do tipo *multilayer perceptron* (MLP) e apresentam arquitetura como a da Figura 2, ou seja, possuem uma camada de entrada para os sinais, uma camada intermediária e uma camada de saída.

O peso da conexão entre o sinal de saída do neurônio i e a entrada do neurônio j é representado por w_{ji} e é calculado na etapa de treinamento. Ao fim dessa etapa, é esperado que a rede forneça respostas razoáveis aos sinais de entradas que não passaram por ela durante o treino [6], ou seja, generalize seus sinais de saída e classifique um sinal entre um gerado pelo WGC ou um WGN.

As entradas para o treino da rede são sinais de tamanho $m = 100$ gerados pelo mapa WGC ou por uma função-amostra de um processo WGN de mesmo tamanho. Os sinais são treinados em ordem aleatória e geram uma saída binária, sendo 0 para WGN e 1 para WGC. Essas saídas por sua vez são comparadas a um gabarito que possui as classificações corretas.

A Rede 1 possui a arquitetura apresentada na Figura 1, essa rede tem m neurônios na camada de entrada, um décimo de m neurônios na camada intermediária e um único na saída. Como a Rede 1 conecta o i -ésimo neurônio da primeira camada da rede, a todos os neurônios da segunda camada, são necessários $m \cdot \frac{m}{10}$ pesos da primeira para segunda camada e mais $\frac{m}{10} \cdot 1$ da segunda para a terceira camada, totalizando $\frac{m}{10}(m + 1)$ pesos. Após a etapa de aprendizado, o número necessário de operações (n_{op}) para classificar um sinal utilizando a Rede 1

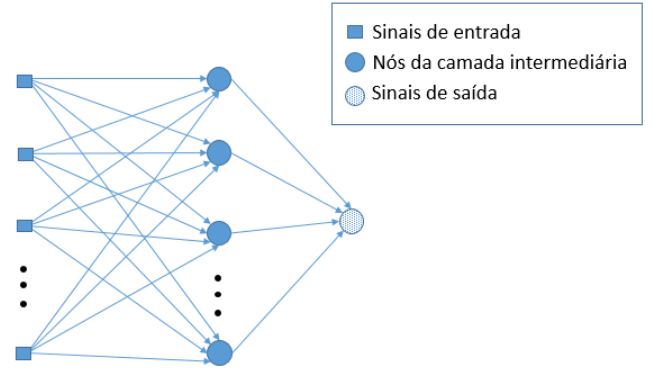


Fig. 2. Estrutura da Rede 1 - Esse rede possui m neurônios de entrada, $m/10$ na camada intermediária e 1 na saída. Os pesos w_{ij} foram omitidos no diagrama.

é de m multiplicações com $m - 1$ adições para cada neurônio da camada intermediária e o mesmo racional para a camada de saída, ou seja $n_{op} = [m + (m - 1)] \cdot \frac{m}{10} + [\frac{m}{10} + (\frac{m}{10} - 1)] = 2m^2 + \frac{m}{10} - 1$.

A Rede 2 possui a estrutura apresentada na Figura 3, essa estrutura é baseada na natureza do sinal caótico, por ele ser determinístico, o ponto $z(n + 1)$ depende do ponto $z(n)$, conforme a Equação (1), assim o i -ésimo neurônio da camada intermediária se conecta aos neurônios i e $i + 1$ da camada de entrada. Assim, teremos m neurônios na camada de entrada, $m - 1$ neurônios na camada intermediária e apenas um na saída. Essa característica, diminui os pesos da Rede 2, pois cada neurônio intermediário terá 2 pesos de entrada e um de saída. Então a quantidade de pesos dessa rede é $2(m - 1) + (m - 1) = 3(m - 1)$. A quantidade de operações dessa rede para classificar um sinal é dada por 2 multiplicações acrescidas de 1 adição para cada um dos $m - 1$ neurônios da camada intermediária, somados a mais $(m - 1)$ multiplicações e $(m - 2)$ adições entre a camada intermediária e o neurônio de saída, sendo $n_{op} = (2 + 1)(m - 1) + (m - 1) + (m - 2) = 5m - 6$.

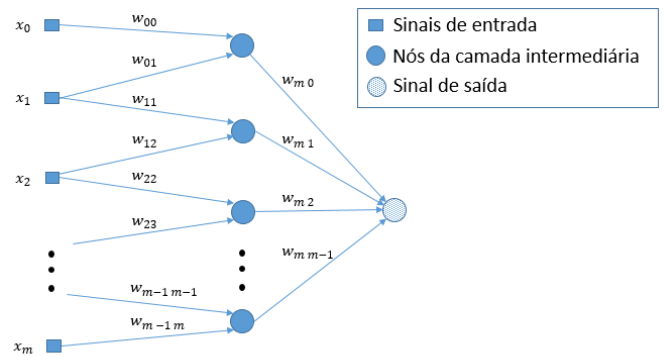


Fig. 3. Estrutura da Rede 2 - Esse rede possui m neurônios de entrada, $m - 1$ na camada intermediária e 1 na saída.

A Rede 3 é apresentada na Figura 4 e possui a mesma estrutura que a Rede 2, porém todos os neurônios da camada intermediária recebem exatamente os mesmos valores de peso em suas conexões. Dessa forma, essa rede possui $2 + (m - 1)$

pesos e faz a mesma quantidade de operações da Rede 2 para classificar um sinal.

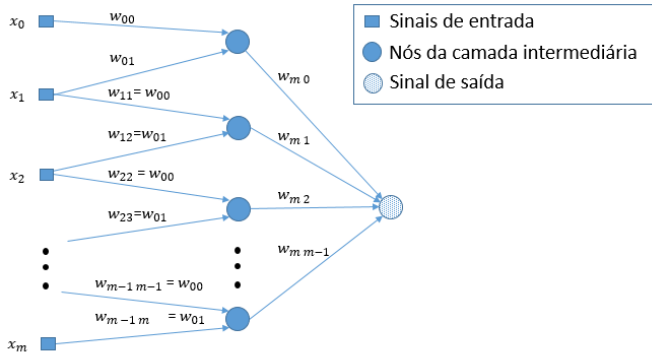


Fig. 4. Estrutura da Rede 3 - Esse rede possui a mesma arquitetura da rede 2, porém todos os neurônios da camada intermediária recebem apenas um par de pesos, sendo eles w_{00} e w_{01} .

IV. RESULTADOS

Essa Seção compara o resultado de classificação das três redes apresentadas. Os pesos w_{ji} dessas redes foram ajustados após uma etapa de treino utilizando o algoritmo de *back-propagation* [6]. A etapa de treino foi realizada com 10^3 sinais de tamanho $m = 100$ pontos, cada sinal foi gerado a partir de condições iniciais aleatórias e treinado a partir de uma taxa de aprendizado $\eta \in [10^{-5}, 10^{-1}]$. A taxa de aprendizado pondera o valor de ajuste dos pesos w_{ji} a cada nova época de treinamento. Esse trabalho utilizou $3 \cdot 10^3$ épocas de treinamento.

Durante a etapa de treino é considerado que a rede aprendeu quando o erro médio quadrático (MSE) de seus resultados diminui em função do aumento das épocas até chegar em um valor de convergência [6]. A média dos valores do MSE em um intervalo i é dada por \overline{MSE}_i , e para esse trabalho foi considerado que houve convergência para as Redes 2 e 3 quando

$$\Delta \overline{MSE}\% = \frac{(\overline{MSE}_{i+1} - \overline{MSE}_i)}{\overline{MSE}_i} \leq 5\% \quad (7)$$

entre dois intervalos consecutivos de 20 épocas. Para a Rede 1 foi necessário um número maior de cenários para se chegar nesse valor de convergência, e o critério foi de 250 cenários, uma vez que a curva de aprendizado possui um comportamento diferente das Redes 1 e 2, conforme Figura 5.

As curvas de aprendizado para as três redes são mostradas na Figura 5 e 6. Na Figura 6 é possível observar que as Redes 2 e 3 convergiram com menos de 100 épocas, enquanto a Rede 1 precisou de 2250. A quantidade de cenários para a convergência de cada Rede, assim como o valor de seu MSE , a quantidade de coeficientes que possui, operações necessárias para se classificar um sinal e seu erro, estão nas Tabelas I e II.

Os dados apresentados na Tabela II mostram que a Rede 1 possui a menor quantidade de erros, porém tem 3,4 e 10 vezes mais coeficientes que as Rede 2 e 3, além de realizar 40,5

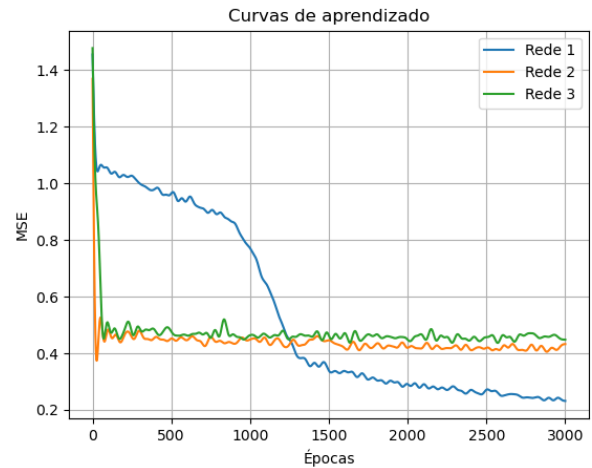


Fig. 5. Curvas de aprendizado das Redes 1, 2 e 3 em função do número de épocas de treino.

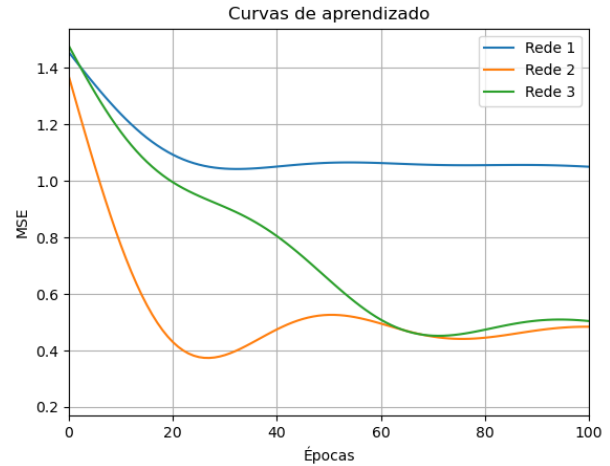


Fig. 6. Curvas de aprendizado das Redes 2 e 3 convergindo em poucas épocas de treino.

vezes o número de operações para classificar um sinal. Quanto a convergência, a Rede 1 demora 56,25 vezes mais para atingir o $\Delta \overline{MSE}\%$ esperado que a Rede 2 e 28,13 vezes mais que a Rede 3. Assim as Redes 2 e 3, que levam em consideração a natureza determinística do sinal gerado pelo WGC, chegam em resultados melhores.

V. CONCLUSÕES E PRÓXIMOS PASSOS

Foi possível classificar um sinal entre gerado pelo mapa WGC ou por um processo WGN utilizando a técnica do perceptron multicamadas. Ainda foi possível adequar a arquitetura da rede para diferenciar a natureza determinística do sinal.

A fim de diminuir as taxas de erro, pretende-se testar outras técnicas, como as árvores de decisão, as *Máquinas de Vetores de Suporte* (SVM) e Métodos de *kernel* [6], além de variar as arquiteturas das redes. Como os sinais WGN utilizados foram gerados a partir de um processo pseudoaleatório, outro passo será substituí-los por sinais gerados em processos realmente aleatórios [7].

TABELA I

TABELA COM AS CARACTERÍSTICAS DE CADA UMA DAS REDES.

Rede	Coefficientes	Operações
1	$0,1m^2 + 0,1m$	$2m^2 + 0,1m - 1$
2	$3m - 3$	$5m - 6$
3	$m + 1$	$5m - 6$

TABELA II

TABELA COM OS RESULTADOS PARA UM CASO ESPECÍFICO COM $m = 100$ NEURÔNIOS.

Rede	Coefficientes	Operações	MSE	Convergência	Erro
1	1010	20009	0,26	2250 épocas	6,6%
2	297	494	0,42	40 épocas	11,6%
3	101	494	0,46	80 épocas	11,7%

AGRADECIMENTOS

ME agradece ao CNPq (311039/2019-7) pelo financiamento parcial desse projeto. O presente trabalho foi realizado com apoio da CAPES - Código de Financiamento 001.

REFERÊNCIAS

- [1] S. Haykin and M. Moher, *Communication Systems*. John Wiley & Sons Inc., 5th ed., 2009.
- [2] K. T. Alligood, T. D. Sauer, and J. A. Yorke, *Chaos*. Textbooks in Mathematical Sciences, Springer, 2000.
- [3] L. H. A. Monteiro, *Sistemas Dinâmicos*. Editora Livraria da Física, 3th ed., 2011.
- [4] M. Eisencraft, L. H. Monteiro, and D. C. Soriano, "White gaussian chaos," *IEEE Communications Letters*, vol. 21, no. 8, p. 1719–1722, 2017.
- [5] B. Lathi, *Sinais e Sistemas Lineares - 2.ed.* Bookman, 2007.
- [6] S. S. Haykin, *Neural networks and learning machines*. Upper Saddle River, NJ: Pearson Education, third ed., 2009.
- [7] P. Peebles, *Probability, Random Variables, and Random Signal Principles*. McGraw-Hill Science/Engineering/Math, 2000.
- [8] J. Proakis, M. Salehi, and G. Bauch, *Contemporary communication systems using MATLAB*. Cengage Learning, 2012.
- [9] M. Koziarski and B. Cyganek, "Image recognition with deep neural networks in presence of noise – dealing with and taking advantage of distortions," *Integrated computer-aided engineering*, vol. 24, no. 4, pp. 337–349, 2017.
- [10] J. Cho, R. Pappagari, P. Kulkarni, J. Villalba, Y. Carmiel, and N. Dehak, "Deep neural networks for emotion recognition combining audio and transcripts," 2019.
- [11] K. Huarng and T. H.-K. Yu, "The application of neural networks to forecast fuzzy time series," *Physica A*, vol. 363, no. 2, pp. 481–491, 2006.